

Smart Cards and PC Cards

Marie Henderson

DSTO-TR-0774

DISTRIBUTION STATEMENT A

Approved for Public Release
Distribution Unlimited

SMART CARDS AND PC CARDS

Marie Henderson

Information Technology Division
Electronics and Surveillance Research Laboratory

DSTO-TR-0774

ABSTRACT

This document introduces both smart cards and PC cards and covers some of their relevant applications to information security. This includes their use in access control, as portable secure storage for cryptographic keys and for computing cryptographic functions. The aim of this document is to highlight the differences between the two card formats (smart card and PC card) and to indicate their respective advantages and disadvantages. The intention is to assist organisations, implementing solutions utilising either format, to select the best option.

APPROVED FOR PUBLIC RELEASE

DEPARTMENT OF DEFENCE
DEFENCE SCIENCE & TECHNOLOGY ORGANISATION

DSTO

19990601140

DTIC QUALITY INSPECTED 4

DSTO-TR-0774

Published by

DSTO Electronics and Surveillance Research Laboratory

PO Box 1500

Salisbury, South Australia, Australia 5108

Telephone: (08) 8259 5555

Facsimile: (08) 8259 6567

© Commonwealth of Australia 1999

AR No. AR-010-836

February 1999

APPROVED FOR PUBLIC RELEASE

Smart Cards and PC Cards

EXECUTIVE SUMMARY

This document introduces both smart cards and PC cards and covers some of their relevant applications to information security. This includes their use in access control, as portable secure storage for cryptographic keys and for computing cryptographic functions. The aim of this document is to highlight the differences between the two card formats and to indicate their respective advantages and disadvantages. The intention is to assist organisations, implementing solutions utilising either format, to select the best option.

The two card formats considered here, smart cards and PC cards, were originally developed for different applications. Smart cards were developed as tamper resistant tokens which could provide certain cryptographic capabilities. They were also developed to replace the magnetic strip card and as such their dimensions are set by the demands of industry conformity and consumer acceptance. PC cards were developed to provide extra storage or to act as interfaces between different computerised devices and are still mainly used for these purposes. However, the PC card also provides a suitable format for applications similar to those of a smart card. Indeed both formats use very similar techniques and technologies to provide these security services.

Smart cards are popular because of their general acceptance, widespread development and cost advantages. As electronic commerce applications increase, the cryptographic capabilities of smart cards will improve and diversify in response to public demand. Nevertheless, the extent of smart card functions will always be limited by the space available. This is also true for PC cards but there is significantly more space available with the PC format. Advances in technologies and techniques will apply to both formats but more of these will fit on a PC card. Indeed, if a high degree of cryptographic capability, more tamper resistant mechanisms and a larger storage space is required, then the PC card format is obviously the better choice. For these reasons it can be expected that these functions will be further developed, in the future, on PC cards as they penetrate the commercial market. It should be noted that smart cards still provide the same, if more limited, functionality at a lower cost.

The capabilities of a smart card may be sufficient in many situations. It is clear that the two formats need to be judged against the requirements of the particular situation and system in which they are to be used. For example, currently marketed smart cards can provide sufficient signature and key exchange capabilities but limited encryption capabilities. In contrast, with the PC cards, both signature and encryption operations can be done as part of the usual card functions. Encryption will always be more viable with a PC card than a smart card because of the differences in input/output capabilities (PC cards have more input/output channels). If physical robustness is required then the PC card may not be as suitable as a smart card in particular environments. From this comparison it is evident that the features required of any system will influence the decision on which format is employed.

DSTO-TR-0774

Authors

Marie Henderson

Information Technology Division

Dr Henderson received a PhD from the University of Queensland in 1997. The work for this degree was carried out within the Department of Computer Science and Electrical Engineering. Since joining the Trusted Computer Systems group at DSTO she has been working in the area of information security.

DSTO-TR-0774

Contents

1	Introduction	1
2	Tokens in Access Control	1
3	Tokens and Public Key Technologies	2
4	Token Types: Smart Cards and PC Cards	3
4.1	An Overview of Smart Cards	3
4.2	An Overview of PC Cards	5
4.3	Smart Cards and PC Cards: First Comparisons	6
5	Card Security	6
5.1	At Production Stage	6
5.2	Memory Access Control	7
5.3	Chip Security	7
5.4	Fortezza Security	8
5.5	Attacks on Cards	8
5.6	Card Expiry	8
6	Cryptographic Functions	9
6.1	Symmetric Key Cryptography	9
6.2	Public Key Cryptography	9
6.3	Hashing	9
6.4	Random Number Generation	10
6.5	Time Stamping	10
6.6	An Example	10
6.7	Fortezza Algorithms	13
6.8	Cryptographically Capable PC cards	15
6.9	Cryptographically Capable Smart Cards	15
6.10	Random Number Generators	15
6.11	Certificates	16
6.12	Key Recovery	17
6.13	Key Generation	17

7	Format Comparisons: Costs, Speeds and Space	18
8	Off-card Encryption	19
9	Conclusions	19
10	Acknowledgements	20
	References	21
	Appendix A	22

Figures

1	Alice sends the encrypted message M to Bob.	11
2	Alice sends the covered message encryption key to Bob.	11
3	Alice sends the signed hash of the message to Bob.	12
4	Bob recovers the key K.	12
5	Bob recovers the message M.	13
6	Bob checks Alice's signature.	13
A1	Alice sends a encrypted and signed message to Bob	22

1 Introduction

Smart Cards and PC cards, formerly known as Personal Computer Memory Card International Association (PCMCIA) cards, are both hardware tokens. The smart cards and PC cards considered in this document perform distinct functions. It is useful to summarise the card systems (smart card and PC card) which will be considered in this document before introducing the two formats. Specifically, the cards must contain a cryptographic engine for computing cryptographic functions (i.e. cryptographically capable) and provide special features aimed at protecting the cards contents (i.e. tamper resistance). The cardholder will identify themselves to the card by using a PIN (Personal Identification Number) or password. This document will consider the following issues:

1. The differences between the two formats, smart cards and PC cards;
2. The cryptographic functions and services performed;
3. The tamper resistance methods employed; and
4. Attacks on the cards.

This information is distributed throughout the document rather than being divided into four separate sections. To assist the reader, a final summary is provided to clarify the main distinctions between the two formats. The investigation begins at the general level of tokens and their use in access control.

2 Tokens in Access Control

Hardware tokens can be used to support access control. There are three general methods that can be applied to provide access control which rely on:

- something the user knows;
- something the user has; or
- something the user is.

Something the user knows normally takes the form of a password or a PIN. Something the user has is more commonly known as a token. A token may be either a badge, a key, a hardware device such as a smart card or even a magnetic strip card. Something the user is relies on recording actual identity through personal characteristics. This can be recorded using biometric technologies which measure either physical characteristics, such as, fingerprints or iris patterns or behavioural characteristics like how someone speaks or writes their signature. These three access control methods can be combined in any one application. For example, a user may gain access to a computer by using a smart card, gain access to their smart card by using a PIN and gain access to the building where the computer is by using a photograph present on the smart card. Using a combination of more than one access control method improves security by making unauthorised access

more difficult to obtain and easier to detect. This document looks specifically at combining tokens with PINs.

Systems that employ tokens combined with PINs have a security advantage over simple password or PIN only systems. Passwords and PINs, by themselves, are not suited to high security applications as people often forget them, write them down or use ones that are easily guessed. Passwords and PINs are also easy to capture through observation. A token provides an additional level of security when combined with a password or a PIN. Users cannot unknowingly share their token and it is evident when a token is lost or stolen. In addition, a user cannot regain the access privileges that their token provided, unless the loss of their token is reported. Because passwords and PINs are easily captured, the passwords or PINs in a token system should be regarded as a secondary identity check and the token as the primary identity check. The security of a token system relies heavily upon the difficulty of obtaining unobstructed access to a token or counterfeiting a token. The disadvantage of a token system is the increase in cost and administration.

The largest security risk in any system comes from people's behaviour. Security procedures can only work when they are adhered to. The best security solutions encourage secure behaviour while detecting insecure behaviour. Tokens provide features that support these goals. Of course, it is still important for any token based system to be implemented within a sound security framework.

3 Tokens and Public Key Technologies

The current development of Public Key Infrastructure (PKI) is mainly due to the global interest in electronic commerce and electronic data interchange. Recently, in Australia, the Project Gatekeeper [1] has put in place an agreed strategy on Public Key Technology for the Commonwealth Government. This will assist in the adoption of inter-operating PKI solutions within government and the wider business community. The Gatekeeper report lists the following tokens as being approved for storage of users private keys:

- floppy diskettes;
- Smart cards;
- PC cards; and
- removable hard disks.

Both PC cards and smart cards can provide a tamper resistant environment for the storage of cryptographic keys (including key for public key cryptography) and the calculation of cryptographic algorithms. With non-tamper resistant or non-cryptographically capable storage then the keys are exposed at a lower security level. It is the tamper resistant storage of cryptographic keys and the provision of a tamper resistant environment for calculation of cryptographic algorithms that is the focus of this document. Some of the cryptographic functions available on these card types are given in Section 6. However, the investigation shall proceed by first looking at the mechanisms employed to provide tamper resistance.

4 Token Types: Smart Cards and PC Cards

The token types considered here are smart cards and PC cards. These two card types have many common elements but are distinctly different in format. These cards rely on a reader to communicate with external devices and perform applications. Card operations are communicated to the cardholder through a display, such as a terminal. In all cases the cardholder must trust the reader and the display. This is an important point, which is relevant to both card systems as an attack may modify the communications between the cardholder and the card reader. Such an attack does not exploit any weakness of the token but rather the overall system structure. Therefore, the level of exposure to this attack depends on the system security rather than the security of the token.

These token types can also be used to perform services in addition to access control. Smart cards and PC cards can provide tamper resistant storage for sensitive data (e.g. PINs) and also cryptographic engines and cryptographic keys. The cryptographic engine may perform a number of cryptographic functions, depending on the specification. For security it is best if the cryptographic engines and keys both reside within the card. Otherwise the keys must leave the secure card environment for execution of cryptographic functions on an external device. This is not an ideal solution as the keys are then exposed at the security level of the external device which may aid key capture. The likelihood and seriousness of such key capture depends on the security of the external device and the possibilities provided to an attacker.

With the provision of cryptographic services a user should also have to verify themselves to the card. This means that capture of the card does not, by itself, allow use of the stored cryptographic keys or access to sensitive data. Currently PINs are widely used to access card functions or data. As biometric techniques advance PINs may be supplemented by other identity checks. Biometrics improve access control as they make it more difficult to share or steal access privileges. However, biometric techniques are still relatively new and pose their own unique set of problems.

4.1 An Overview of Smart Cards

A smart card is effectively a plastic card with a micro-circuit (chip) embedded within the card. Smart cards rely on VLSI (Very Large Scale Integration) chip technology for information processing and storage. The purpose of a smart card is to provide a secure and tamper resistant module for the information storage and processing. To do this the smart card chip:

1. utilises a secure file access system;
2. computes cryptographic functions; and
3. defends against illegal access attempts.

These aspects of a smart card will be discussed in subsequent sections. Various features of a smart card have been standardised by the International Standards Organisation (ISO). ISO 7816 includes standards for the physical card characteristics, electronic signals and

transmission protocols. The dimensions of a smart card are set to those of a normal credit card, 86mm long by 54mm wide by 0.75mm thick. Lower limits for card strength have also been set. For example, a card must work correctly and must not have any cracking after 1000 bendings where either:

- the long side is bent to a 2cm deep curve with 30 bendings per minute; and
- the short side is bent to a 1cm deep curve with 30 bendings per minute.

There are other minimal strength requirements that include tolerance of shock and environmental stress. These standards are designed to ensure that smart cards operate sufficiently in normal day-to-day use. Parts of the standards are still evolving but they do not cover the size or performance of the chip. The size of the chip is influenced by the reliability and robustness required. Evidently, a large chip is more likely to fail, even in ordinary use, where considerable amounts of bending and shock can be expected. A variety of industry standards also exist which have complicated the standardisation process.

The smart card communicates with the reader through eight contacts. These contacts are conducting surfaces that are linked to the smart card chip using thin wires. The functions of six of the cards contacts have been standardised, the other two have been set aside for future developments. Only one of these six contacts is used for input and output between the smart card and the reader. Full duplex smart cards (one input contact and one output contact) are currently being discussed. The single Input/Output (I/O) construction has implications for the speed of smart card I/O operations. Specifically, this I/O construction has the potential to create an I/O data bottleneck.

The contacts of a smart card reader must connect with those of the smart card before transmission can begin between the reader and the smart card. The smart card reader may use either sliding contacts or landing contacts. With sliding contacts the smart card is pushed into place under the reader's contacts. With landing contacts the smart card is placed within the reader and then the reader's contacts close onto the smart cards. There are advantages and disadvantages with either system. A good connection is more likely to be achieved with sliding contacts. However, using sliding contacts requires greater insertion pressure and wears the smart card contacts faster. With landing contacts it is easier to ensure that the power is removed from the smart card as soon as it is withdrawn from the reader. This ensures that all smart card contacts are properly reset. The best smart card readers combine a small amount of sliding with a landing contact system, thereby providing the better features of both reader types.

As smart cards become more widely accepted their price falls and the diversity of cards available increases. Devices using smart card technology now come in many different formats and with a variety of features. For example, there are contactless cards (which use a small embedded antenna to communicate with the reader) and key shaped cards (which also provide the usual physical access security). Another interesting development is super smart cards. They have the same functionality as a smart card but include a keyboard, LCD display and battery and are similar in size to a pocket calculator. In this case the reader and display that the cardholder must trust are combined with the token. This provides a higher level of security as this card avoids the capture of the cardholder's PIN, display of false information or fraudulent verification of an abandoned process.

4.2 An Overview of PC Cards

PC cards were developed as memory cards for data transfer and to act as interfaces between different computerised instruments. Aspects of PC cards have been standardised by the PCMCIA. The types available, PCI, PCII and PCIII depend on the thickness of the card but each is 85.6mm long by 54mm wide. The thicknesses are 3.3mm, 5.0mm and 10.5mm, respectively. PC cards can contain a number of chips and perform a variety of functions. They are more rigid than a smart card but also have to conform to various reliability conditions. Lower limits for the mechanical and environmental tolerances of PC cards have been set in the PCMCIA standards. The weight of a PC card would depend on the components placed within the card.

It is practical to develop tamper resistant modules for the PC card format. Smart cards were originally developed to replace the magnetic strip card and as such needed to be the same size for public acceptance. There are no technical reasons why chips, like those in smart cards, cannot be placed within a PC card. Indeed, PC cards are now being manufactured offering similar capabilities to smart cards (see points 1, 2 and 3 above in Section 4.1). These PC cards also utilise VLSI chip technologies but due to their rigidity, more than one chip may be placed on a card. The best known examples were developed as part of the Fortezza program which was created by the NSA to help provide secure electronic messaging. Significantly, the first tokens to contain the Fortezza cryptographic algorithms were PC cards. Some of the Fortezza algorithms are now available on smart cards and in software. Some of the Fortezza algorithms were originally classified as secret but these have since been declassified making the software versions possible. The Fortezza program did not specify which type of PC card was to be used.

Tamper resistant, cryptographically capable PC cards have also been developed outside of the Fortezza program. They provide cryptographic services similar to those supplied by smart cards. The availability of these cards will increase with the expected increase in applications and demand for security services. Many devices, such as laptops, already contain PC card readers (for other non-security related applications). Therefore, the obvious choice is to use PC cards with these devices. Note that PC cards can also act as readers for smart cards. Also, PC cards have more space available which means higher security and increased functionality can be provided as compared to smart cards. This should motivate further development of cryptographically capable PC cards. PC cards communicate through sockets (on the card) and pins (in the reader) rather than contacts (a useful analogy may be that of the normal 2 or 3 pin powerpoint and plug connection). These pins are physical components of the reader and should not be confused with card access PINs. The standard requires that cards function up to a minimum of 10 000 insertions in an office environment and 5 000 insertions in a harsh environment. The functions of the contact sockets and pins have been standardised which means it is possible for each of these card types to use a single reader. However, not all card readers are built to accommodate all card thicknesses. There are 68 sockets on each card, of which 16 are available for I/O. As in the case of smart card contacts, the I/O sockets and pins work in both directions (both as input and output).

4.3 Smart Cards and PC Cards: First Comparisons

The main difference between the smart card and PC card formats is in the space available within the cards and the I/O mechanisms employed. The dimensions of smart cards and PC cards differ only in their thickness. PC cards do not have to be as flexible as a smart card and so more of the card space can be used for chips containing memory or cryptographic functions and utilities such as monitoring or shielding devices aimed at improving tamper resistance. With a smart card the single I/O is not well suited for processing large amounts of data. The 16-bit parallel I/O connections of PC cards can cope with higher data rates but may wear faster and be easier to damage.

Smart cards are the obvious choice to replace credit cards as they improve security while retaining the same consumer accepted shape. If an application requires high functionality, tamper resistance or I/O throughput then the PC card format is the obvious choice because, in regards to these features, a PC card is at least as capable as a smart card. This is a general statement that is independent of any future technological advances but, of course, dependent on actual implementations.

5 Card Security

This section introduces some of the security techniques applied to, or available with, smart cards and PC cards. This section does not include the security of the cryptographic algorithms, which form part of the card's functions, but rather the methods used to secure a card's contents. PC cards and smart cards can utilise similar techniques to secure card contents. Card systems can use both logical and physical security techniques. Cheaper smart cards or PC cards would employ fewer of the techniques described below. Each technique would have an economic and performance cost associated with it. Certain security features may have drawbacks in that they make the card more fragile and therefore less reliable. This is because of the following fact: if a security device is triggered then the card will respond accordingly, regardless of whether the trigger was the result of an attack or an accident. Space requirements could also limit the number of security features present on any one card format, although it would be possible to fit more features within a PC card because of the increased space available. It is important to realise that security methods are being continually developed in response to successful public attacks.

5.1 At Production Stage

The chips in a smart card or PC card are regarded as being impossible to counterfeit. Therefore great care is taken to protect cards from being stolen or from being used if stolen during the manufacture and personalisation stages. Each card contains unique information that identifies the card. During production a serial number is stored on the card. This information is permanently fixed by blowing fuses which access these areas. During manufacture, various test contacts and testing modes are required to analyse the chip. These are disabled once testing has ceased, again by blowing fuses.

5.2 Memory Access Control

A key feature of security cards (smart cards and PC cards) is that all access to memory is controlled by the microprocessor. Memory management circuits also provide hardware protection against unauthorised access. Data stored in a card's memory is logically divided into four categories:

1. free read and write access;
2. restricted access where data can be transmitted from the card provided access has been given by the card controlling program (for example, with PIN checking);
3. forbidden access where data can only be used within the card's programs (used for storing cryptographic keys and the PIN); and
4. manufacturer's fabrication area which cannot be written to (the card's serial number is placed here).

Memory can consist of many different memory types (ROM, RAM, etc.). The important point is that there is no access to memory that is not controlled by the card's operating system (also known as the mask).

5.3 Chip Security

Chips can themselves be protected by several means. The glue used to bond the chip to the card frame is stronger than the chip, causing the chip to be destroyed in a removal attempt. The card can be protected against PIN experimentation by counting the number of incorrect PINs entered and locking the card if this exceeds a pre-set limit (this number would normally be reset to zero once a valid PIN is entered). The chip itself is designed to make internal analysis difficult. Internal analysis includes examination with an electron microscope or circuit analysis. Features of the chip design may include some of the following security enhancing techniques.

- Layering information so that functions or data are spread across layers as opposed to being arranged logically. This includes burying ROM (memory that holds the cards operating system) at the lowest level and bus and address scrambling.
- Using a metal casing to shield the chip which cannot be removed without destroying the chip.
- Incorporating dummy components.
- Including detectors for electron radiation (so if radiation is detected sensitive data can be destroyed), temperature and low or high frequency inputs (for circuit analysis and reliability).
- Scattering oscillators throughout the chip to shield against electromagnetic examination.

As mentioned not all of these techniques may be available on any particular card. Some may cause unacceptable operational faults if they are prone to being triggered by a false alarm.

5.4 Fortezza Security

Fortezza cards were designed to secure classified cryptographic algorithms and so may apply techniques not publicly available. The algorithms are contained within a single on-board chip known as the Capstone chip. Memory and other functions residing outside of the Capstone chip are likely to be secured in ways similar to smart cards, although other means are possibly applied. Keys for encrypting data on Fortezza cards are kept in volatile memory. If the card is removed from the reader or power is lost to the card then these keys are erased. This helps to prevent anyone else from accessing these keys.

5.5 Attacks on Cards

Due to their similar design and applications, attacks relate to both card types. As mentioned, with tamper resistant modules it is important to recognise that they are only tamper resistant and not tamper proof. This issue is not related to the security of the cryptographic algorithms placed on a card but on the ability of the cards to keep their contents secure. It is not possible to prove, in a mathematical sense, that a card is secure as it is not possible to demonstrate that no failure modes exist in which card security is compromised. The best a card can do is to increase the cost of any attack and delay success. The best design would:

- make the cost prohibitive when compared to the reward obtained from compromising a card; and
- delay the success of an attack until the information obtained was no longer useful.

There are many attacks on cards but most require obtaining the card. Attacks can use methods to induce computational errors to recover cryptographic keys, through to special techniques to invade cards or even modify cards. Some attacks are non-invasive and so change the focus of the security to include the systems in which the cards are used. For example, attacks could employ modified readers. Such attacks would be difficult and expensive to mount but are possible. Other attacks, that rely on obtaining the card, are still important where any knowledge obtained from a single card can be applied to others. This is most important when cryptographic keys are shared. Such knowledge could even be used to speed up future attacks by using the determined layout or faults of a particular manufacturer's card. For examples of attacks on cards see references [2-4] below.

5.6 Card Expiry

An expiry date for cards helps to keep track of cards, ensures that they are replaced before they begin to malfunction and provides a means of updating card technologies. Expired

or failed cards should be properly disposed of and secured against theft as they may still contain information useful to an attacker.

6 Cryptographic Functions

Cryptography enables parties to securely exchange information by preventing access to any unauthorised party. This is achieved by scrambling the information (encryption) using certain cryptographic functions so that only the intended recipients can recover the original information (decryption). The cryptographic services that either smart cards or PC cards can perform are not set by any standard. These depend on the envisaged applications and security requirements as balanced by the economic and performance costs. There are a number of cryptographic services that can be combined on either card type with those relevant to this discussion outlined below. The number of cryptographic services available on a card is limited by the card size and limits of software and hardware. There are many different cryptographic algorithms which perform these services, not all of which are suited to the smart card or PC card format. Both card types may also include other special purpose units such as a cryptographic coprocessor to speed up the calculation of cryptographic functions. Simple descriptions are provided for the convenience of the reader.

6.1 Symmetric Key Cryptography

The main use is to provide data confidentiality. Two users who wish to communicate securely, using Symmetric key cryptography, must share a common key, called a secret key, which is used to both encrypt and decrypt. Symmetric key cryptography can provide efficient encryption and decryption of data but also significant key management problems as both the sender and the receiver must have a copy of the secret key.

6.2 Public Key Cryptography

Each user has their own private key, which they keep secret, and a public key, which is made available to every other user in the system. Public key systems do not require communicating parties to exchange keys. Public key systems are much slower than symmetric key systems so they are normally used with smaller amounts of data. Common applications include exchanging secret keys between two parties (for symmetric key cryptography) and creating digital signatures for data. Digital signatures can provide authentication (they supply proof of the sender's identity), non-repudiation (the sender cannot deny their signature) and data integrity (ensures the data has not been altered en route).

6.3 Hashing

This is involved with digital signatures and data integrity. As public key cryptography is relatively slow it is better to sign a smaller hashed version of your data. A hash function

is one-way so the data cannot be regained from the hash value. At the receiver's end, the signed hash value can be checked against the hash value calculated by the receiver.

6.4 Random Number Generation

Some cryptographic schemes require the input of a random number. It is important that the random number generator used is sufficiently good, as attacks can take advantage of a poor random number generator.

6.5 Time Stamping

Time stamps provide timeliness and uniqueness to data. For example, a signature on some data may have a validity period which can be checked against the data's time stamp and a replay attack will be exposed (a replay attack involves re-sending a captured message to produce some known outcome). Time stamps require a trusted time source.

6.6 An Example

An example is included below to demonstrate how these cryptographic functions operate and interact. The particular scenario considered is a simplistic scheme but includes most of the cryptographic functions listed above. The purpose of this example is for demonstration only and should not be considered as a cryptographic mandate. The diagrams below represent each of the processes performed. A complete diagram depicting how these processes may be combined is included in Appendix A. The diagram in Appendix A demonstrates the flow of processes from Alice to Bob.

Example

Suppose that Alice wishes to send a signed and encrypted message to Bob and that they have agreed on the symmetric key algorithm, hash function, key exchange algorithm and signature algorithm to be used. In this example we use the following conventions.

- The symmetric key algorithm is **S** and the symmetric key used is **K**.
- The hash function is **H**.
- The key exchange algorithm is **Ep** for encrypting and **Dp** for decrypting. The public keys of Alice and Bob for this algorithm are **PEA** and **PEB** respectively and the corresponding private keys are **PDA** and **PDB** respectively. The **PDA** key is only known to Alice and the **PDB** key is only known to Bob.
- The signature algorithm is **Es** for encrypting and **Ds** for decrypting. The public keys of Alice and Bob for this algorithm are **SEA** and **SEB** respectively and the corresponding private keys are **SDA** and **SDB** respectively. The **SDA** key is only known to Alice and the **SDB** key is only known to Bob.

Steps that Alice performs

Alice has composed a message M which she wishes to send to Bob. Alice generates the key K using the supplied random number generator and encrypts the message M using S and then sends this to Bob, see Figure 1.

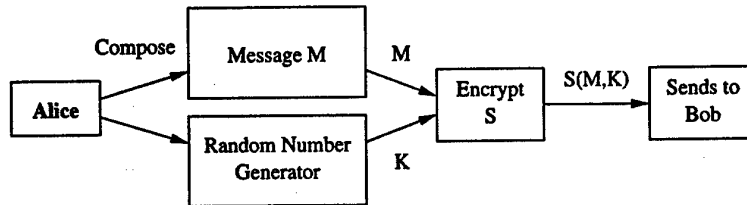


Figure 1: Alice sends the encrypted message M to Bob.

Bob will need the key K to recover the message so Alice encrypts K with E_p and Bob's public key exchange key, PEB , and sends this to Bob as well, see Figure 2 (The key K comes from a previous step which is indicated by placing K in a circle rather than a box).

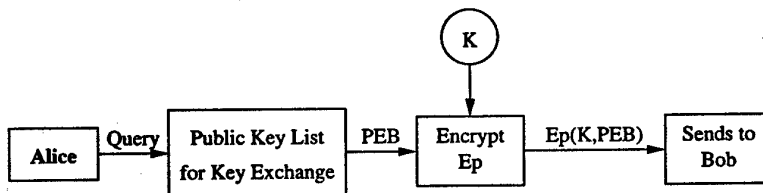


Figure 2: Alice sends the covered message encryption key to Bob.

Now Bob will be able to recover K and the message M but he cannot be sure that the message came from Alice. Alice can provide assurance by sending Bob a signed version of her message M . However, it is more efficient to sign a hash of the message M as this is, in general, a lot smaller than the message. In our example Alice hashes the message M and then signs this hash value and sends this to Bob, see Figure 3.

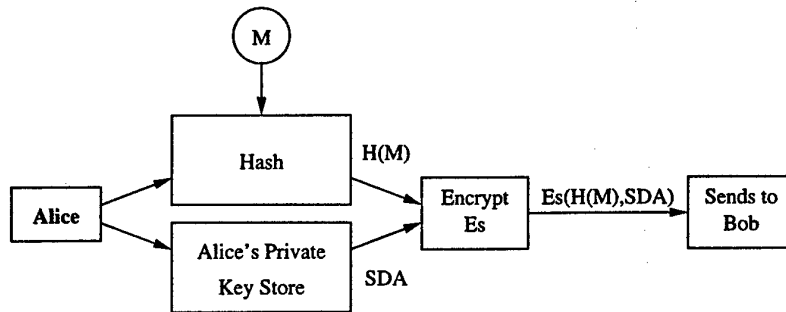


Figure 3: Alice sends the signed hash of the message to Bob.

The separation of the signing, message encryption and key encryption functions is made clear by the diagrams. Indeed, it is possible to perform each of these functions separately or individually as the application requires. For example, Alice may wish to only apply a signature to an message which has not been encrypted. The order of application of functions could also vary between applications. For example, the signed hash of the message may be encrypted along with the message. In this case the Message M in Figure 1 would be a combination of the message that Alice wishes to send Bob (say M1) and the signed hash value of this message ($Es(H(M1,SDA))$) which is determined as in Figure 3. Not all variations along these lines are appropriate and the variant used will depend on the system policy and the application. The reader should be aware that the scenario selected here (where the signature is attached but not encrypted) is only one of many possible constructions.

Steps that Bob performs

Bob now has the encrypted message, $S(M,K)$, the wrapped key K , $Ep(K,PEB)$, and the signed hash of the message M, $Es(H(M),SDA)$. To recover the message Bob will need to use the key K . To recover the key K Bob uses his private key exchange key PDB (Figure 4).

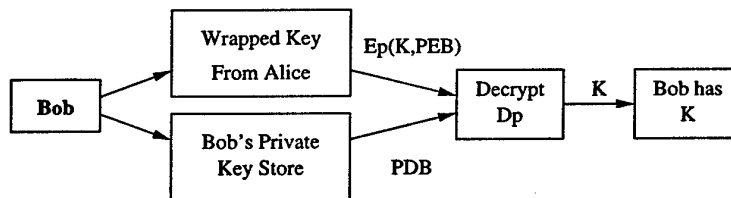


Figure 4: Bob recovers the key K.

Bob now has the key K which he can use to recover the message M (Figure 5).

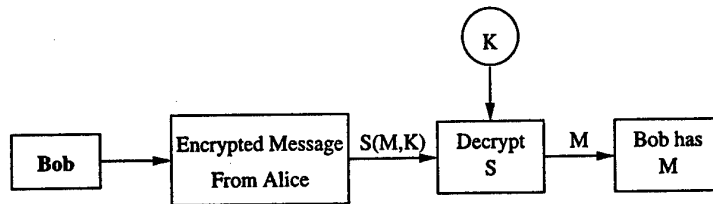


Figure 5: Bob recovers the message M .

Bob now has the message M but to be sure it did originate from Alice he must check Alice's signature (Figure 6).

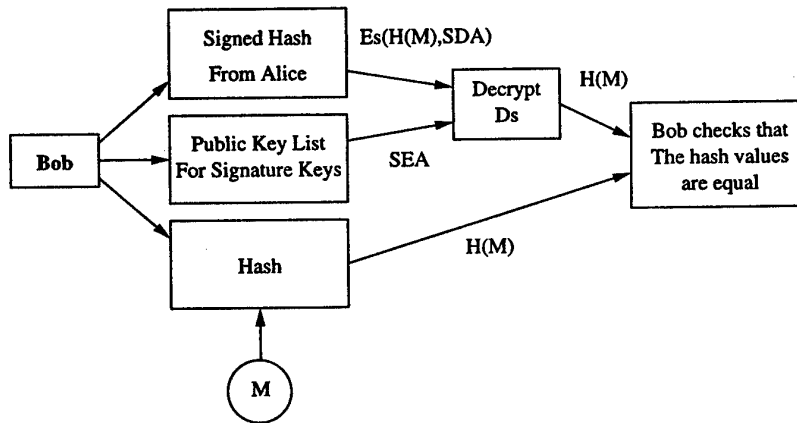


Figure 6: Bob checks Alice's signature.

Bob now has the message M and has checked that the two hash values match. This guarantees that the message has not been altered en route. Bob can be sure that the message came from Alice as she is the only person who knows SDA and therefore is the only person who could have produced the correct signed hash value. Anyone who has the message and the signed hash value could verify, in the same way as Bob has, that the message originated from Alice.

6.7 Fortezza Algorithms

The Fortezza algorithms were originally contained on the Capstone chip. The Capstone chip is placed within a PC card along with other components. This chip was designed to use the same key-escrow features as the Clipper chip. The Clipper chip key-escrow features have been publicly discussed. The key-escrow can now be switched off but there are still reasons to use a key-recovery mechanism in some circumstances. This is discussed in a Section 6.12. The Fortezza functions are listed below:

- the Skipjack algorithm, a symmetric key cryptosystem;

- KEA, a key exchange algorithm;
- DSA, a digital signature algorithm;
- SHA-1, a version of the hash function known as the Secure Hash Algorithm;
- a high-speed general purpose exponentiation algorithm;
- a general purpose random number generator that uses a pure noise source; and
- a real-time tamper resistant clock for time stamping.

As already mentioned some of the Fortezza algorithms were originally classified. The Skipjack and KEA algorithms were declassified on the 24th of June 1998. The security of these algorithms was not supposed to rely on the secrecy of the algorithms and this does seem to be the case (although public investigations are on-going). The specifications for the Skipjack and KEA algorithms is currently available on the world wide web. Oddly, these algorithms are still covered by U.S. export policy with an export license required to export Fortezza cards and Fortezza enabled software applications. The explanation is that there are different government departments dealing with each issue and currently all cryptographic exports from the U.S. are restricted regardless of their classification status.

The motivation for this declassification is to make software and smart card versions of the algorithms available for the U.S. Defense Message System (DMS) program and to encourage their commercial development and implementation. In a NSA press release, concerning this declassification, it is stated that the algorithms (Skipjack and KEA) are not intended to become Advanced Encryption Standard (AES) candidates. The final sentence of the NSA press release has interesting implications: "Software Fortezza is a transition vehicle in migrating to AES based commercial security solutions for the Defense Information Infrastructure". It should be mentioned that the discussions on AES are still at the development stage. The AES meetings are at present still considering candidate algorithms. It may take 5 years or more before the AES algorithm is available in hardware tokens. It is not even clear, at this stage, whether the AES will ever become available on cryptographically capable tokens. The AES is not meant to replace all current encryption algorithms but to provide an additional choice and has come about because of the recognition that DES is approaching the end of its useful lifetime. Apart from all this, Skipjack is still intended for widespread use within the US DoD in the medium term. Other Fortezza algorithms, KEA, DSA and SHA-1 are expected to be universally supported in both the DoD and the wider community in general.

There are a few companies advertising Fortezza enabled products. The NSA catalogue of commercial products [5] lists four companies with Fortezza enabled products. They are Group Technologies Corporation, Mykotronx, National Semiconductor and Spyrys. The catalogue includes descriptions of the products and some links to company pages. In addition, Spyrys supply some Fortezza algorithms on a smart card and in software. Litronic [6] also supply Fortezza PC cards and software Fortezza. It is possible that there are other suppliers.

6.8 Cryptographically Capable PC cards

There are cryptographically capable PC cards, other than those implementing Fortezza functions, currently available. In 1995, the PCMCIA established a working group dedicated to standardising and promoting the PC card as a security device. The number of security PC cards should increase as security applications increase and diversify. The cryptographic functions available, as such, will reflect those available on smart cards. As already mentioned, with either format the functions and algorithms implemented are not restricted by any standard.

6.9 Cryptographically Capable Smart Cards

A wide range of cryptographic algorithms are implemented on current smart cards. The list below presents some of the algorithms implemented on currently marketed smart cards. This list is not claimed to be complete or accurate (abilities change with each new release) and is only intended to indicate the variety of algorithms being implemented on smart cards.

Symmetric Key: DES or triple DES.

Public Key: RSA ¹, GQ Zero-knowledge, Elliptic curve (provides similar security to RSA and DSA but with a shorter key length), DSA ² and Diffie-Hellman (for key exchange).

Hash Functions: MD2, MD5 and SHA-1.

Smart cards may include special purpose high speed coprocessors and random number generators. The features present on a smart card are only restricted by physical robustness requirements of ISO 7816.

6.10 Random Number Generators

If keys are to be generated on either card type then a method of generating random numbers is required. The security of these algorithms is therefore affected by the quality (or real randomness) of the numbers generated. Hardware random number generators are easier to design and generally better than software ones, although both rely on random input (see [7]). Generators that use oscillators and capacitors can be built into VLSI chips such as the ones used in PC cards and smart cards. These can be enclosed within the tamper resistant module containing the chip so that they are also shielded from attack. Various statistical tests exist that can be used to gauge the effectiveness of a random number generator.

Some Fortezza family PC Cards do actually use a random number generator that relies a pure noise source. It should also be possible for a smart card to use similar types of

¹Some cards can perform 1024-bit or 2048-bit RSA. These key lengths are currently considered to be long enough for long term security. The 1024-bit RSA public-keys can be generated by the card.

²The key length is limited, by Federal Information Processing Standard FIPS-186, to the range of 512 to 1024-bits: 512 bits is not regarded as secure for the long-term but is the only version currently implemented outside the Fortezza program

random number generators, although it is not clear if this has actually been done. As hardware based generators rely on some physical phenomena it is possible that the output is, to a degree, biased or correlated. De-skewing techniques can be applied to create true randomness from a natural but biased or correlated random source. Although real random number generators can be constructed in this way, PC card and smart card companies do not state whether any de-skewing techniques are applied with their random number generators.

6.11 Certificates

With any public key cryptosystem it is necessary to make the public keys available to the people you wish to communicate with. Information identifying users and their public keys can be distributed in a data structure known as a certificate. A particular user's certificate contains information identifying them, like their name and organisation, along with their public key. It is crucial that this public key is uniquely linked to the correct owner in a trusted way. To provide this assurance, certificates include information about the issuing authority who also signs each certificate they issue. The issuing authorities are successive, with the authorities at one level being certified by those at a higher level. Of course, this hierarchy must have some limit. Authorities at the highest levels use different policies to cross authenticate each other.

Some PC cards and smart cards have dedicated memory space for certificates. Fortezza cards have well developed certificate policies because of their intended use in messaging. The Fortezza program stipulates the use of the X.509 version of a certificate within its own certificate structure. There is a set amount of space for certificates on Fortezza cards (the Spyru Fortezza Crypto card has space for 27 certificate and the Spyru Lynks Privacy Card has space for 50 certificates). One certificate space, on a Fortezza card, is designated for the certificate of the highest authority in the user's certificate domain (all other certificates generated within the domain may then be checked by back tracking down from this one). Only the site security officer (SSO) has the authority to write to this certificate space. This authority is granted by the cards logical security policies and therefore it would be possible to construct other secured certificate slots if required in special circumstances. Keeping this certificate on-card provides a higher level of assurance as the certificate is less exposed to tamper-based attacks. Other certificate spaces can be accessed by either the user or SSO.

It would be possible to construct similarly secured certificate slots on other PC cards or smart cards. The number of certificate spaces is limited on a smart card because of the restrictions placed on memory space available. Currently it is only possible to store 2-4 certificates on a smart card (based on having 8K of memory space). The number that may be placed on a Fortezza card is much larger. It is unclear whether there are any substantial security advantages of having many certificates on-card. In [8] it is stated that certificates can be stored locally (off the card) but that for some applications only the card's certificate storage space should be used because of its hardware based access controls. This would include the certificate of the highest certifying authority of the user's domain (to avoid tampering), but may not be necessary (for security reasons) for any other certificates. The user must trust the validity of the highest authorities certificate

but all other certificates within their domain can be checked against this one. Reasons of ease of use and portability would support storage of other certificates on the token.

6.12 Key Recovery

There are good reasons, in certain circumstances, to use key recovery with cryptographically capable tokens. If the card functions are used to secure data (messages, documents etc.) then this data will be unrecoverable if the card is lost or malfunctions. It may be necessary to provide another way of generating the keys or to keep a separate copy of the keys. If this facility is required then it is easiest to secure data using a per data key *S*, secure each *S* with a public key and then store the data along with the encrypted key *S*. A copy of the private key corresponding to the public key (used to encrypt *S*) is held by the key-recovery authority. In this way a different key (*S*) can be used for each set of data but only one key (the private key) needs to be stored off card for recovery. If a copy of the key used to secure data is to be kept by another authority then this should be different from any signature key, otherwise the non-repudiation service of the digital signature system would be degraded.

6.13 Key Generation

Symmetric keys would normally be created by cards to avoid key management problems. It is possible for symmetric keys to be distributed by a trusted source but this complicates key management and requires secure key distribution. With public key pairs it is possible to either generate the keys on the card, which is referred to as decentralised key generation, or have a trusted authority generate the key pairs and distribute them to the user or place them on the user's card, which is referred to as centralised key generation. The following points are relevant.

1. Centralised key generation can provide stronger security because better techniques may be used to generate and test keys.
2. Centralised key generation offers an easy means of performing key recovery because the keys may be stored, if required, when they are generated.
3. Centralised generation costs more because a trusted party is required to generate and distribute keys and creates a possible threat to end users (as they must trust the party generating the keys).
4. With centralised key generation certificates containing public key material can be generated at the same time as the keys whereas with decentralised key generation each user needs to generate their key and then send this with a certificate application to the certificate authority (this could result in a delay between key generation and certificate generation). The keys and certificate application need to be bound cryptographically for assurance.

These key generation techniques can be applied individually to the different cryptographic algorithms provided by any one card. As mentioned it is expected that symmetric keys

are generated on the card. Public keys may be generated in either way depending on their intended use. For non-repudiation it is best if signature keys are generated on the card. With public keys, centralised generation is more appropriate (although not essential), when key recovery is required.

7 Format Comparisons: Costs, Speeds and Space

The following 1994 baselines for smart cards was extracted from [9] and [10].

1994 Baselines for Smart Cards

1. 8 bit, 3.5-8 MHz microprocessor (16-32 bit at 20 MHz to become available).
2. Non-volatile memory 8-16 Kbytes for data storage.
3. ROM 8-16 Kbytes for card operating system.
4. RAM 256-512 bytes for operating system computations.
5. EEPROM 2-8 Kbytes externally accessible by user, non-volatile.
6. Cost estimates: \$7-30/PKI cards (according to the functionality and security measures required) in 1000-5000 lots and \$150-200/reader in 100-500 lots.

For comparison we include information on the PC card developed as a result of the Fortezza program. These details are taken from the SpyruS home page, accessible from [5].

Fortezza PC card estimates

1. A 32-bit, 20-40 MHz processor and an ARM 60 processor.
2. Volatile memory of 64 Kbytes.
3. Non-volatile memory of 128-512 Kbytes.
4. Cost: \$80/card and \$150-300/reader.
5. Battery life of 7 years.

From these estimates it is seen that there is substantially more memory available on a Fortezza PC card than with a smart card. This has implications for the amount of data that can be placed on either card type (for example, certificates, keys, PINs, etc.) and their respective computational capabilities.

8 Off-card Encryption

Note that, due to I/O speed, it may be impractical to use a symmetric key contained within a smart card to secure large amounts of data or to frequently encrypt data. An alternative is to generate the symmetric key within the smart card and then use off-card software to encrypt the data. As the data is present on the system where the software resides then the data can only be regarded as secure as the system it has been generated on anyway. A possible problem can arise if the secret keys released to the system can be analysed to gain other information. For example, suppose an attacker has broken the cards random number generator by observing the symmetric keys that are generated and used off-card. The attacker can then use this knowledge to attack other algorithms (such as DSA, which uses random numbers). This attack could also be mounted by a message receiver even when the symmetric key encryption is performed on card (the receiver of a message must recover the symmetric key to decrypt the message). The number of symmetric keys available to this receiver based attack is the number of encrypted messages that the sender has actually sent them and is probably far less than the number available to the off card encryption attack. These attacks rely on exploiting a weakness in the random number generator.

Even with a Fortezza PC card it may not be optimal to do all calculations on the card. It is suggested in [8] that for performance sensitive applications, slower card functions (such as SHA-1) could be performed off-card. Obviously, the PC cards performance is still influenced by the I/O performance, although this is better than for smart cards. It seems that encryption with Fortezza is performed on-card for reasons other than to just avoid secret key capture. In [11] the author states that encryption and digital signatures are performed on-card as it is necessary to 'tie the encryption to key exchange since the federal government escrows the master keys'. They conclude that if key escrow is not required then encryption is more likely to be performed off-card as in the smart card situation. An advantage of performing off-card encryption is the flexibility provided: any available software encryption algorithm can be used (provided an appropriate key can be generated).

9 Conclusions

The two formats considered here, smart cards and PC cards, were originally developed for different applications. Smart cards were developed as tamper resistant tokens which could provide certain cryptographic capabilities. They were also developed to replace the magnetic strip card and as such their dimensions are set by the demands of industry conformity and consumer acceptance. PC cards were developed to provide extra storage or to act as interfaces between different computerised devices and are still mainly used for these purposes. However, the PC card also provides a suitable format for applications similar to those of a smart card. Indeed both formats use very similar techniques and technologies to provide these security services.

Smart cards are popular because of their general acceptance, widespread development and cost advantages. As electronic commerce applications increase, the cryptographic capabilities of smart cards will improve in response to public demand. Nevertheless, the

extent of smart card functions will always be limited by the space available. This is also true for PC cards but there is significantly more space available with the PC format. Advances in technologies and techniques will apply to both formats but more of these will fit on a PC card. Indeed, if a high degree of cryptographic capability, more tamper resistant mechanisms and a larger storage space is required, then the PC card format is obviously the better choice. For these reasons it can be expected that these functions will be further developed, in the future, on PC cards as they penetrate the commercial market. It should be noted that smart cards still provide the same, if more limited, functionality at a lower cost. The capabilities of a smart card may be sufficient in many situations. It is clear that the two formats need to be judged against the requirements of the particular situations and systems in which they are to be used. For example, currently marketed smart cards can provide sufficient signature and key exchange (although KEA is not yet implemented) capabilities but limited encryption capabilities. In contrast, with the Fortezza PC cards, both signature and encryption operations were expected to be normal functions. Encryption will always be more viable with a PC card than a smart card because of the differences in I/O capabilities. It is evident that the features required of any system will influence the decision on which format is employed.

10 Acknowledgements

The references [12-15] were also used to gather general material for this document. I would like to thank Dr. M. Ozols and Dr. N. Parker for their comments and suggestions and Dr. B. Mahony and D. J. McCarthy for their help with the figures.

References

1. Office of Government Information Technology, *Gatekeeper*, 1998
<http://www.ogit.gov.au/>
2. R. Anderson and M. Kahn, *Tamper Resistance - a Cautionary Note*, The second USENIX Workshop on Electronic Commerce Proceedings, Oakland, California, pp 18-21, November 1996.
3. E. Biham and A. Shamir, *Differential Fault Analysis of Secret Key Cryptosystems*, Lecture Notes in Computer Science, Advances in Cryptology, Proceedings of CRYPTO'97, pp 513-525, 1997.
4. P. Kocher, J. Jaffe and B. Jun, *Differential Power Analysis*,
<http://www.cryptography.com/dpa/index.html>
5. NSA Catalogue, <http://www.nsa.gov:8080/program/missi/cat.index.html>
6. Litronic Fortezza solutions,
<http://www.Litronic.com/solutions/fortezza.html>
7. D. Eastlake, S. Crocker and J. Schiller, *Randomness recommendations for Security*,
<http://sunsite.auc.dk/RFC/rfc/rfc1750.html>
8. Spyrus, *Fortezza Application Implementors Guide*, 1996.
9. R. Merckling and A. Anderson, *Smart Card Introduction*,
<http://www.geocities.com/ResearchTriangle/Lab/1578/rfc570.txt>
10. N. Tesch, *Smart Card Technology*,
<http://komar.cs.stthomas.edu/qm425/tesch3.htm>
11. T. Parrish, *Fortezza - The Pentagon Plugs into PCMCIA Based Data Encryption*, Defense Electronics, Vol 27, no. 6, 1995.
12. PCMCIA home page, <http://www.pc-card.com/>
13. A. J. Menezes, P. C. van Oorschot and S. A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997.
14. W. Caelli, D. Longley and M. Shain, *Information Security Handbook*, Macmillan Press, 1994.
15. M. Hendry, *Smart Card Security and Applications*, Artech House, 1997.

Appendix A

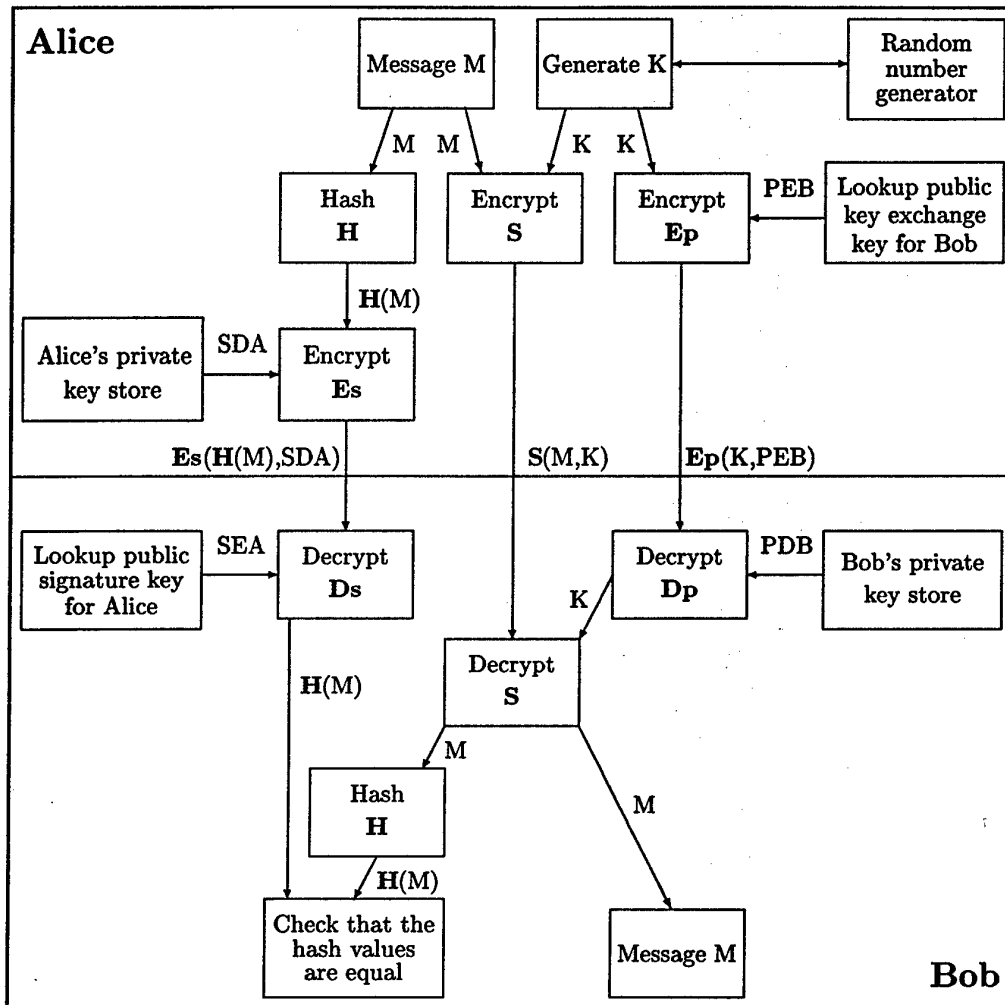


Figure A1: Alice sends a encrypted and signed message to Bob

DISTRIBUTION LIST

Smart Cards and PC Cards

Marie Henderson

Number of Copies

DEFENCE ORGANISATION

S&T Program

Chief Defence Scientist	}	
FAS Science Policy	}	1
AS Science Corporate Management	}	
Director General Science Policy Development		1
Counsellor, Defence Science, London		Doc Data Sht
Counsellor, Defence Science, Washington		Doc Data Sht
Director General Scientific Advisers and Trials	}	
Scientific Adviser - Policy and Command	}	Doc Data Sht
Navy Scientific Adviser		Doc Data Sht
Scientific Advisor - Army		Doc Data Sht
Air Force Scientific Adviser		1
Director Trials		1

Aeronautical & Maritime Research Laboratory

Director, Aeronautical and Maritime Research Laboratory	1
---	---

Electronics & Surveillance Laboratory

Director	1
Chief Information Technology Division	1
Research Leader Command and Control and Intelligence Systems	1
Research Leader Military Computing Systems	1
Research Leader Command, Control and Communications	1
Executive Officer, Information Technology Division	Doc Data Sht
Head, Information Architectures Group	1
Head, Information Warfare Studies Group	Doc Data Sht
Head, Software Systems Engineering Group	Doc Data Sht
Head, Year 2000 Project	Doc Data Sht
Head, Trusted Computer Systems Group	1
Head, Advanced Computer Capabilities Group	Doc Data Sht
Head, Systems Simulation and Assessment Group	Doc Data Sht
Head, C3I Operational Analysis Group	Doc Data Sht
Head, Information Management and Fusion Group	1

Head, Human Systems Interaction Group	Doc Data Sht
Head, C2 Australian Theatre	1
Head, Information Architectures Group	1
Head, Distributed Systems Group	Doc Data Sht
Head, C3I Systems Concepts Group	1
Head, Organisational Change Group	Doc Data Sht
Head, Crypto Mathematics Research Group	1
Dr Weimin Zhang	1
Publications and Publicity Officer, ITD	1
DSTO Library and Archives	
Library Fishermens Bend	1
Library Maribyrnong	1
Library Salisbury	2
Australian Archives	1
Library, MOD, Pyrmont	Doc Data Sht
Capability Development Division	
Director General Maritime Development	Doc Data Sht
Director General Land Development	Doc Data Sht
Director General C3I Development	1
Navy	
SO(Science), Director of Naval Warfare, Maritime Headquarters Annex, Garden Island	Doc Data Sht
Army	
ABCA Office, G-1-34, Russell Offices, Canberra	4
SO(Science), DJFHQ(L), Milpo, Enoggera, Qld 4057	Doc Data Sht
NAPOC QWG Engineer NBCD c/- DENGERS-A, HQ Engineer Centre Liverpool Military Area, NSW 2174	Doc Data Sht
Intelligence Program	
DGSTA, Defence Intelligence Organisation	1
ASINFOSEC, Defence Signals Directorate	1
Library, Defence Signals Directorate	Doc Data Sht
Defence Security Branch	
ASSEC	1
Acquisition Program	
DGCOMMS Defence Acquisition Organisation	1

Corporate Support Program(libraries)

Officer in Charge, TRS, Defence Regional Library, Canberra	1
Officer in Charge, Document Exchange Center	Doc Data Sht
National Library of Australia	1
Additional copies for DEC for exchange agreements	
US Defense Technical Information Center	2
UK Defence Research Information Centre	2
Canada Defence Scientific Information Service	1
NZ Defence Information Centre	1

UNIVERSITIES AND COLLEGES

Australian Defence Force Academy Library	1
Head of Aerospace and Mechanical Engineering, ADFA	1
Deakin University Library, Serials Section (M List)	1
Senior Librarian, Hargrave Library, Monash University	1
Librarian, Flinders University	1

OTHER ORGANISATIONS

NASA (Canberra)	1
Australian Government Publishing Service	1
The State Library of South Australia	1
Parliamentary Library of South Australia	1

ABSTRACTING AND INFORMATION ORGANISATIONS

INSPEC: Acquisitions Section Institution of Electrical Engineers	1
Library, Chemical Abstracts Reference Service	1
Engineering Societies Library, US	1
Materials Information, Cambridge Science Abstracts, US	1
Documents Librarian, The Center for Research Libraries, US	1

INFORMATION EXCHANGE AGREEMENT PARTNERS

Acquisitions Unit, Science Reference and Information Service, UK	1
Library - Exchange Desk, National Institute of Standards and Technology, US	1

Spares

DSTO Salisbury, Research Library	10
----------------------------------	----

Total number of copies: 67

Page classification: UNCLASSIFIED

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION DOCUMENT CONTROL DATA				1. CAVEAT/PRIVACY MARKING Not Applicable	
2. TITLE Smart Cards and PC Cards			3. SECURITY CLASSIFICATION Document (U) Title (U) Abstract (U)		
4. AUTHOR(S) Marie Henderson			5. CORPORATE AUTHOR Electronics and Surveillance Research Laboratory PO Box 1500 Salisbury, South Australia, Australia 5108		
6a. DSTO NUMBER DSTO-TR-0774		6b. AR NUMBER AR-010-836		6c. TYPE OF REPORT Technical Report	
7. DOCUMENT DATE February 1999					
8. FILE NUMBER N 9505-17-15		9. TASK NUMBER JNT 96/178		10. SPONSOR DG C3ID	
11. No OF PAGES 33		12. No OF REFS 15			
13. DOWNGRADING / DELIMITING INSTRUCTIONS Not Applicable			14. RELEASE AUTHORITY Chief, Information Technology Division		
15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT <i>Approved For Public Release</i> <small>OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE CENTRE, DIS NETWORK OFFICE, DEPT OF DEFENCE, CAMPBELL PARK OFFICES, CANBERRA, ACT 2600</small>					
16. DELIBERATE ANNOUNCEMENT No Limitations					
17. CITATION IN OTHER DOCUMENTS No Limitations					
18. DEFTEST DESCRIPTORS Magnetic cards Technology innovation Computer information security Cryptology					
19. ABSTRACT This document introduces both smart cards and PC cards and covers some of their relevant applications to information security. This includes their use in access control, as portable secure storage for cryptographic keys and for computing cryptographic functions. The aim of this document is to highlight the differences between the two card formats (smart card and PC card) and to indicate their respective advantages and disadvantages. The intention is to assist organisations, implementing solutions utilising either format, to select the best option.					

Page classification: UNCLASSIFIED